NPS-53Jy74122

NAVAL POSTGRADUATE SCHOOL

Monterey, California

The Performance Characteristics of Some Reliability Growth Models

T. Jayachandran and L. R. Moore, III

December 1974

Technical Report for Period January 1974-December 1974

Approved for public release; distribution unlimited

pared for:
ef of Naval Research, Arlington, Virginia 22217

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral Isham Linder
Superintendent

Jack R. Borsting
Provost

The work reported herein was supported in part by the Foundation Research Program of the Naval Postgraduate School with funds provided by the Chief of Naval Research.

This report was prepared by:

L. D. KOVACH, Acting Chairman
Department of Mathematics

ROBERT R. FOSSUM
Dean of Research

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: NPS-53Jy74122	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): The Performance Characteristics of Some Reliability Growth Models	5. TYPE OF REPORT & PERIOD COVERED: Technical Report Jan. 1974/Dec. 1974
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): T. Jayachandran and L. R. Moore, III	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61152N; RR 000-01-10; WR 5 - 0001
11. CONTROLLING OFFICE NAME AND ADDRESS: Chief of Naval Research Arlington, Virginia 22217	12. REPORT DATE: December 1974
	13. NUMBER OF PAGES: 35
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Reliability, Growth models, failure rate, mission reliability exponential distribution

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A reliability growth model is an analytical model that accounts for changes in reliability due to design changes and other corrective actions taken during the development and testing phases of a reliability program. This paper describes the results of a Monte Carlo study comparing the performance characteristics of four reliability growth models that have been proposed in the reliability literature.

SUMMARY

A reliability growth model is an analytical model that accounts for changes in reliability due to design changes and other corrective actions taken during the development and testing phases of a reliability program. This paper describes the results of a Monte Carlo study comparing the performance characteristics of four reliability growth models that have been proposed in the reliability literature.

1. INTRODUCTION

A reliability growth model is an analytical model that accounts for changes in reliability (usually an improvement) due to design changes and other corrective actions taken during the devolopment and testing phases of a reliability program. A good reliability growth model would be useful in determining a test plan that will lead to the development of an acceptable product within a reasonable time period. Once a test plan is developed the model may be used to monitor the progress of the reliability program and if warranted, initiate changes such as accelerated testing. At any point in time during the test phase an idea as to how much additional testing would be required to achieve a reliability goal may be obtained using the model.

There are several reliability growth models available in the literature. Some of these models are probabilistic in nature i.e., the models do not allow for incorporation of information obtained during the test phase. The second type of models, generally known as statistical models, involve unknown parameters that can be estimated using available test data. This paper discribes the results of a study comparing the performance for four statistical models. Two of these models are intended for use with time to failure data. The other two models use bernoulli type data; a "success" is taken to mean "the satisfactory operation for a specified period of time" and the specified time period is often referred to as "mission time".

A description of the reliability growth models selected for this study is given in Section 2. The Monte Carlo procedure used is described in Section 3 and Section 4 contains the methods of estimating the parameters of the different models. The results of the study are found in Section 5. Corcoran and Read [3] conducted a Monte Carlo study to compare several reliability growth models that includes two of the models considered in this paper. However, there are several differences in the methodology used in the two studies.

2. Description of the Reliability Growth Models.

Model I: Duane [4] developed this model based on his analysis of large quantities of data on various types of airborne equipment. If T is the accumulated test time on a given piece of equipment, and λ_T is the instantaneous failure rate at time T then, Duane's model states that

$$\lambda_T = \beta(1-\alpha)T^{-\alpha}$$

where β and α are unknown parameters.

Model II: Suppose that (i) items are put on test sequentially and after each failure an appropriate corrective action is taken and (ii) the time to failure on the i^{th} test has an exponential distribution with parameter λ_i . Then, Weiss [5] assumes that the parameters λ_i satisfy the equation

$$\lambda_i = \frac{B + i}{Ai}$$

for some unknown parameters A and B .

Model III: Let

$$p_r = P[\text{"mission success" after the } r^{th} \text{ failure has been observed and appropriate corrective action has been taken}]$$

Chernoff and Woods [2] model p_r as an increasing function of r defined by

$$p_{r+1} = 1 - e^{-(\alpha+\beta r)}$$

Model IV: Wolman [6] developed this model assuming that

(i) testing is conducted sequentially and each trial results in a "mission success" or a "mission failure"

(ii) on each trial one of two types of failures can occur; the first type is an inherent or non-correctable failure and the second type of failure is a transient failure that can be eliminated by corrective action and

(iii) whenever a test results in a "failure" an attempt is made to eliminate the failure mode and the attempt may result in a) either a complete elimination b) a partial elimination or c) no effect on the failure mode.

Let

r = probability of an inherent failure, constant from test to test

q = probability of a transient failure at the beginning of the test program

β_i = factor by which the corrective action after the i^{th} failure, reduces the probability of a transient failure

$\alpha_i = \prod_{j=0}^{i-1} \beta_j$ the cumulative factor by which the initial probability of a transient failure is reduced as a result of the corrective actions after each of the i failures and

$\beta_{n+1}(i)$ = probability of a "mission success" on the (n+1)st test if i failures are observed in the first n tests

The Wolman model assumes that

$$p_{n+1}(i) = 1 - r - q\alpha_i$$

Wolman points out that in many situations a good approximation is obtained by assuming that the improvement factors β_i are all equal to a constant value β. In this case, the model equation reduces to

$$p_{n+1}(i) = 1 - r - q\beta^i$$

The latter version with $\beta = .2$, .5 and .8 is used in this study.

3. Monte Carlo Procedure

In order to compare the growth models described in the previous section, five decreasing sequences of λ (failure rate) values are selected. Each sequence consists of 16 values ranging between .70 and .05; this represents a growth in reliability, assuming that all times are measured in mission units, from an initial value of .50 to a final value of .95 approximately. The growth patterns represented in these sequences vary from a very rapid growth to a very slow growth.

For each of the 16 λ's in a given sequence 10 exponential failure times t_{ij} $i=1,2,\ldots,16$; $j=1,2,\ldots,10$ are generated. Since a reliability growth model is to predict the λ values that are in practical situations unknown, the generated data is used to first estimate the λ values in a given sequence in two ways. In the first method the t_{ij}'s are treated as the observed exponential failure times and in the second case the t_{ij}'s are converted into bernoulli data by noting whether each t_{ij} excedes 1 or not. This is done to study the effect of different data types on the prediction capabilities of the growth models. The estimated λ values are used to empirically determine the unknown parameters of the growth models using least squares techniques. To account for statistical variability the whole process described above is replicated 100 times resulting in 100 empirical growth curves for each of the λ sequences and each of the reliability growth models.

If λ_i , $i = 1,2,\ldots, 16$ are the failure rates in a given sequence 100 predicted values λ_i^* are obtained from the empirical curves for each of the models. The "goodness" of a reliability growth model is now measured using the following three measures:

$$M_1 = \sum_{i=1}^{16} \left| \frac{\lambda_i - \overline{\lambda_i^*}}{\lambda_i} \right|$$

$$M_2 = \sum_{i=1}^{16} \left[\frac{\lambda_i - \overline{\lambda_i^*}}{\lambda_i} \right]^2$$

$$M_3 = \max_i \left| \frac{\lambda_i - \overline{\lambda_i^*}}{\lambda} \right|$$

where $\overline{\lambda_i^*}$ is the average of the 100 predicted values λ_i^* . All three measures are relative measures i.e., each difference $\lambda_i - \overline{\lambda_i^*}$ is divided by the true value λ_i in computing the measures. The reason for choosing relative measures is that it would be important for a growth model to predict the small λ values at the end of a sequence rather well. In other words, these measures attach more weight to the differences $\lambda_i - \overline{\lambda}_i^*$ at the end of each decreasing sequence.

4. Estimation of Parameters

As indicated earlier the generated failure times t_{ij}, $j = 1,2,\ldots, 10$ are used to estimate the failure rates λ_i in two different ways. For the case where the t_{ij}'s are used directly the estimator of λ_i is the maximum likelihood estimator $\hat{\lambda}_i = 1/\overline{t}_i$ where $\overline{t}_i = \frac{1}{10} \sum_{j=1}^{10} t_{ij}$. Since models III and IV describe the growth in p_i the probability of a mission success, $\hat{\lambda}_i$ is converted into an estimator of p_i by noting that

$$
\begin{aligned}
p_i &= P[\text{mission success on } i^{th} \text{ trial}] \\
&= P[\, t_{ij} \geq 1] = e^{-\lambda_i}
\end{aligned}
$$

Thus, the maximum likelihood estimator of p_i is $\hat{p}_i = e^{-\hat{\lambda}_i}$. In the second case, the t_{ij}'s are transformed to bernoulli data and the λ_i and p_i are estimated from the transformed data as follows:

Let

$$
x_{ij} \begin{cases} 1 & \text{if} \quad t_{ij} \geq 1 \\ 0 & \text{if} \quad t_{ij} < 1 \end{cases}
$$

and let $Y_i = \sum_{j=1}^{k} X_{ij}$. Then, Y_i has a binomial distribution with parameters p_i and k . The maximum likelihood estimators of p_i and λ_i based on Y_i are $\hat{\hat{p}}_i = \frac{Y_i}{k}$ and $\hat{\hat{\lambda}}_i = -\ln \hat{\hat{p}}_i$ respectively. The methods for obtaining the empirical growth curves and the predicted λ values from the growth curves are now described. Since these methods are the same for both sets of estimators $\hat{\lambda}_i$, $\hat{p}_i$ and $\hat{\hat{\lambda}}_i$, $\hat{\hat{p}}_i$ the symbols $\tilde{\lambda}_i$, $\tilde{p}_i$ will be used to denote either of the two sets of estimators. Also, the symbols λ_i^* , p_i^* denote the λ_i , p_i values as predicted by the empirical growth curves.

For model I the equation describing the relationship between the failure rate λ_T and the total test time T is

$$\lambda_T = \beta(1-\alpha)T^{-\alpha}$$

or equivalently

$$\ln \lambda_T = \ln[\beta(1-\alpha)] - \alpha \ln T$$

The parameters α and β are estimated by replacing λ_T with $\tilde{\lambda}_i$ and T with $T_i = \sum_{\ell=1}^{i} \sum_{j=1}^{10} t_{\ell j}$ in the latter equation and then applying standard regression methods. Once the least squares estimators $\hat{\alpha}$ and $\hat{\beta}$ are obtained, the predicted values λ_i^* of the failure rates λ_i are given by

$$\lambda_i^* = \hat{\beta}(1 - \hat{\alpha})T_i^{-\hat{\alpha}} \qquad i = 1,2,\ldots,16$$

The parameters A and B of model II are estimated using regression methods on the relationship

$$\tilde{\lambda}_i = \frac{B}{A}\frac{1}{i} + \frac{1}{A} \qquad i = 1,2,\ldots,16$$

and the predicted values are

$$\lambda_i^* = \frac{\hat{B} + i}{\hat{A}i} \qquad i = 1,2,\ldots,16$$

For model III the regression equation has the form

$$-\ell n(1 - \tilde{p}_i) = \alpha + \beta(i-1) \qquad i = 1,2,3,\ldots,16$$

and for model IV the equation is

$$1 - p_i = r + q\beta^{i-1} \qquad i = 1,2,\ldots,16 \qquad \text{with } \beta \text{ known} \quad .$$

The rest of the procedure is the same as for models I and II.

5. SIMULATION RESULTS

The results of the simulation are presented in tables 1.1 to 1.5. The five tables correspond to the five sets of 16 λ values used in generating the failure times. These λ values are listed at the top of the tables. As indicated earlier the five sets chosen represent reliability growth ranging between a slow growth to a rapid growth. Also, the selection of the λ's in these sets are made so as to generally follow the growth indicated by the models that are being tested. The entries in the tables are the observed values of the three measures M_1, M_2, M_3 for each of the four models with three cases under model IV.

It may be observed from tables 1.1 to 1.5 that with time to failure data model I performed better than the others in all cases except one (table 1.4) where model III was best. For attributes data model IV with $\beta = .8$ appears to outperform the other models. This result coincides with a similar conclusion by Bresenham [1].

In order to study the case where exactly one item is put on test at any time, the simulation was rerun generating only one failure time t_i instead of 10 failure times t_{ij} for each of the 16 λ's in a given sequence. There was a difference in the generation of attribute data from the failure times. Let X_i denote the largest integer less than or equal to t_i, the time to failure on the i^{th} trial with failure rate λ_i, $i = 1,2,\ldots,16$. Since the t_i are assumed to be measured in mission units X_i is equivalent to the number of full missions completed before the occurrence of a failure. Thus, X_i has a geometric distribution with parameter p_i, the probability of a mission success on the i^{th} trial. The maximum likelihood estimator of p_i is $\hat{p}_i = 1/(X_i + 1)$. Except for this difference in the way the p_i's are estimated the rest of the procedure is the same as before. The results of this second study are in tables 2.1 to 2.5. In this case, model III appears to do uniformly well with time to failure data and with attributes data model III and model IV with $\beta = .8$ seems to perform better than the others.

On the basis of this study, the following general conclusions may be drawn:

(1) with time to failure data and several items on test simultaneously, model I appears to predict reliability growth accurately

(2) with attributes data and several items on test model IV with $\beta = .8$ performs well

(3) with time to failure data and one item on test model III is a good model to use and

(4) with attributes data and one item on test either model III or model IV with $\beta = .8$ may be used.

Graphs comparing true growth curves (curves down through the λ values used in the study) with the curves estimated by model I using time data and model IV using attribute data with $\beta = .8$ are presented in Figures 1.1 to 1.5. Figures 2.1 to 2.5 compare the true growth curves with the curves estimated by model III using time and attribute data and model IV using attribute data with $\beta = .8$.

The computer programs used in the study are attached.

TABLE 1.1

True Values of Lambda: .702 .434 .320 .255 .213 .183 .161 .144 .130 .118 .109 .101 .0936 .0876 .0823 .0776

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	0.69	0.04	0.10	9.46	6.23	0.99
Model II	2.45	0.65	0.40	2.28	0.54	0.38
Model III	2.27	0.44	0.37	7.91	4.66	.778
Model IV β = (.2)	7.49	5.16	1.13	6.70	4.04	0.99
Model IV β = (.5)	4.52	2.13	0.79	4.09	1.57	0.67
Model IV β = (.8)	1.50	0.24	0.23	1.55	0.18	0.16

TABLE 1.2

True Values of Lambda: .700 .573 .508 .466 .435 .411 .392 .376 .362 .350 .340 .330 .322 .314 .307 .301

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	0.68	0.04	0.11	2.69	0.50	0.22
Model II	1.67	0.27	0.23	1.35	0.18	0.20
Model III	1.04	0.09	0.15	2.76	0.58	0.32
Model IV β = (.2)	2.67	0.58	0.33	2.21	0.40	0.25
Model IV β = (.5)	1.85	0.32	0.27	1.58	0.20	0.18
Model IV β = (.8)	1.09	0.08	0.11	0.39	0.01	0.06

TABLE 1.3

True Values of Lambda: .700 .353 .238 .180 .145 .122 .106

.0933 .0837 .0760 .0697 .0644 .0600 .0562 .0529 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	0.78	0.06	0.11	11.00	8.04	0.82
Model II	1.61	0.16	0.11	0.92	0.06	0.08
Model III	3.17	0.86	0.50	10.20	6.94	0.86
Model IV $\beta = (.2)$	8.42	6.57	1.26	7.33	4.88	1.08
Model IV $\beta = (.5)$	4.22	1.93	0.76	3.59	1.28	0.61
Model IV $\beta = (.8)$	3.87	1.16	0.43	3.81	1.13	0.44

TABLE 1.4

True Values of Lambda: .700 .564 .460 .379 .315 .263 .221 .186 .157 .132 .112 .0953 .0810 .0689 .0587 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	1.88	0.48	0.55	11.60	13.80	2.91
Model II	8.32	8.93	1.85	8.01	8.16	1.78
Model III	0.56	0.02	0.06	11.90	19.80	3.85
Model IV $\beta = (.2)$	14.10	24.30	3.04	12.90	20.10	2.77
Model IV $\beta = (.5)$	10.60	14.20	2.39	9.75	11.40	2.15
Model IV $\beta = (.8)$	2.45	0.92	0.65	1.95	0.47	0.48

TABLE 1.5

True Values of Lambda: .700 .456 .315 .227 .170 .132 .106 .0880 .0757 .0672 .0613 .0572 .0544 .0524 .0510 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	1.23	0.13	0.14	11.80	9.74	1.54
Model II	3.67	1.07	0.38	3.73	1.11	0.39
Model III	3.70	1.09	0.41	9.88	6.60	0.88
Model IV $\beta = (.2)$	11.80	12.10	1.47	10.80	10.20	1.34
Model IV $\beta = (.5)$	6.57	3.93	0.88	6.01	3.11	0.78
Model IV $\beta = (.8)$	3.68	1.12	0.47	3.47	1.07	0.53

TABLE 2.1

True Values of Lambda: .702 .434 .320 .255 .213 .183 .161 .144 .130 .118 .109 .101 .0936 .0876 .0823 .0776

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	19.20	43.10	5.44	5.71	6.15	2.21
Model II	13.20	11.30	10.80	30.7	83.8	4.83
Model III	9.63	6.37	0.76	4.56	1.68	.559
Model IV β = (.2)	40.50	297.00	15.40	16.50	85.00	6.26
Model IV β = (.5)	36.70	229.00	13.50	13.50	63.30	7.74
Model IV β = (.8)	23.00	35.00	35.00	3.88	1.37	0.57

TABLE 2.2

True Values of Lambda: .700 .573 .508 .466 .435 .411 .392 .376 .362 .350 .340 .330 .322 .314 .307 .301

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	18.60	38.40	5.02	3.73	3.51	1.78
Model II	118.00	877.00	8.35	8.68	14.10	3.15
Model III	6.36	2.70	0.49	2.16	.354	.240
Model IV $\beta = (.2)$	24.60	169.00	12.60	7.80	24.40	4.86
Model IV $\beta = (.5)$	24.40	164.00	12.40	7.27	23.10	4.75
Model IV $\beta = (.8)$	13.80	12.70	1.61	3.83	3.52	1.76

TABLE 2.3

True Values of Lambda: .700 .353 .238 .180 .145 .122 .106

.0933 .0837 .0760 .0697 .0644 .0600 .0562 .0529 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	19.20	41.20	5.20	5.47	6.17	2.26
Model II	127.00	1010.00	8.68	50.80	216.00	6.40
Model III	11.10	8.53	0.93	4.79	1.76	0.64
Model IV $\beta = (.2)$	45.80	303.00	14.60	15.90	60.50	7.30
Model IV $\beta = (.5)$	41.40	247.00	13.40	10.40	29.50	5.12
Model IV $\beta = (.8)$	25.90	44.50	2.12	3.54	.834	0.29

TABLE 2.4

True Values of Lambda: .700 .564 .460 .379 .315 .263 .221

.186 .157 .132 .112 .0953 .0810 .0689 .0587 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	21.00	70.00	7.49	8.13	11.90	2.91
Model II	182.00	2670.00	25.50	27.00	77.50	5.02
Model III	9.91	7.21	12.90	4.67	2.46	0.93
Model IV $\beta = (.2)$	49.20	433.00	17.40	25.90	167.00	11.90
Model IV $\beta = (.5)$	47.50	475.00	19.30	23.30	157.00	11.80
Model IV $\beta = (.8)$	28.40	81.60	5.72	8.54	10.10	1.91

TABLE 2.5

True Values of Lambda: .700 .456 .315 .227 .170 .132 .106

.0880 .0757 .0672 .0613 .0572 .0544 .0524 .0510 .0500

MODEL NUMBER	TIME DATA			ATTRIBUTE DATA		
	M1	M2	M3	M1	M2	M3
Model I	19.50	45.10	5.62	6.08	6.46	2.27
Model II	145.00	1370.00	11.10	48.50	205.00	6.34
Model III	10.90	9.22	1.11	5.50	2.21	0.59
Model IV β = (.2)	52.20	421.00	17.40	25.10	204.00	13.90
Model IV β = (.5)	46.30	353.00	16.50	18.20	124.00	10.90
Model IV β = (.8)	25.90	45.70	2.06	3.42	0.90	0.35

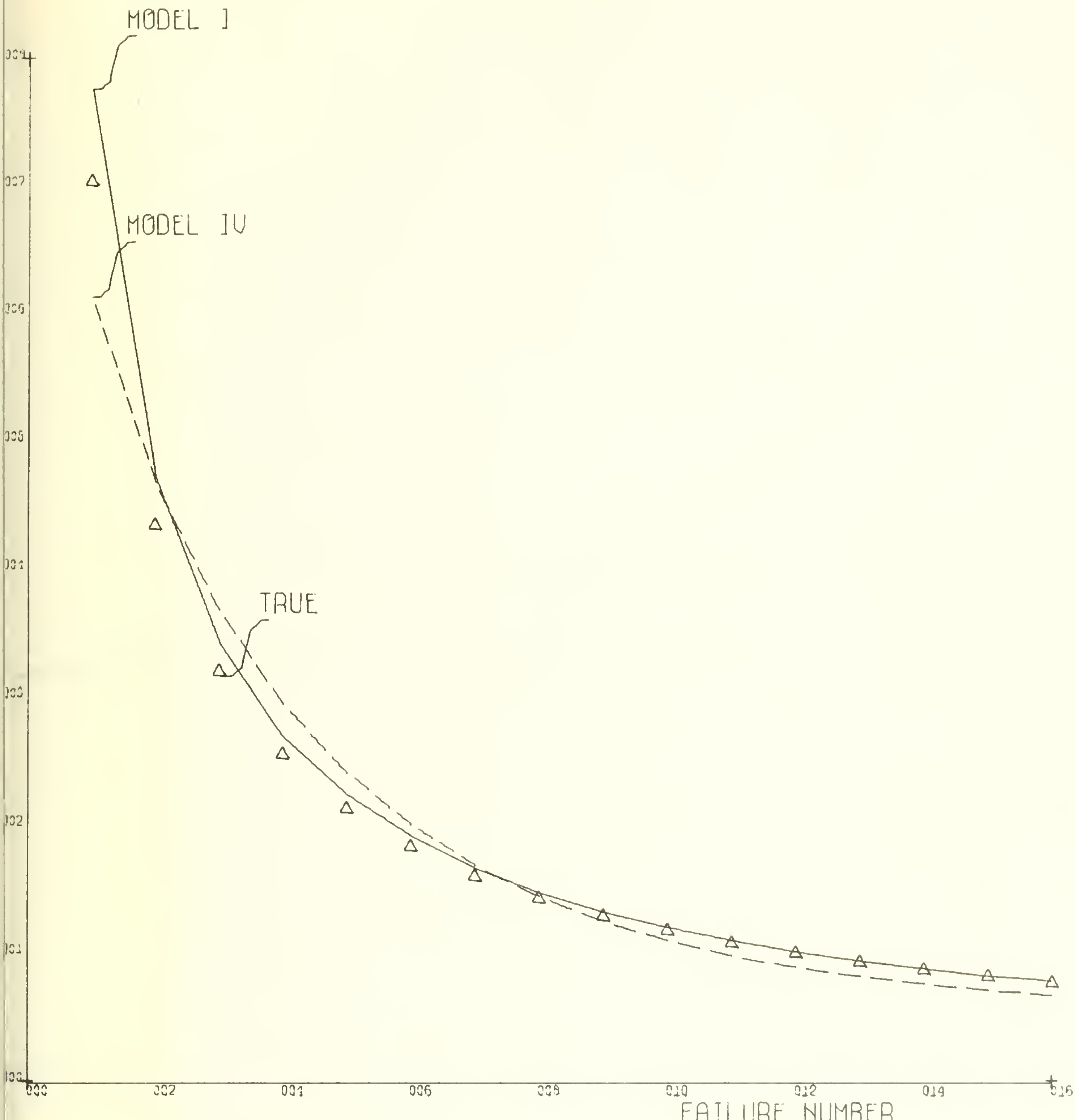

FIGURE 1.1 GRAPH OF PREDICTED FAILURE RATES FOR MODEL I, MODEL IV(.8), AND TRUE RATE

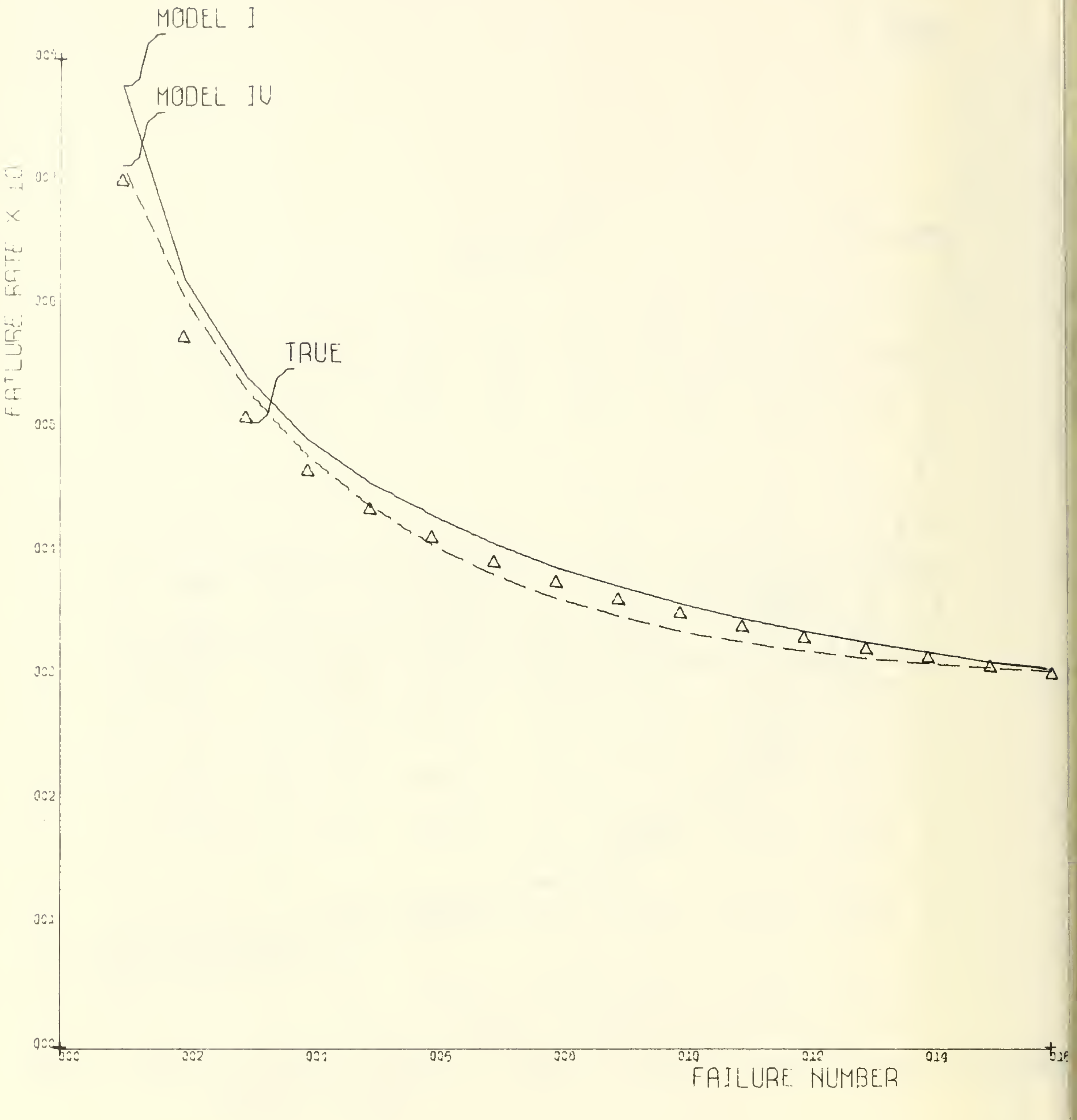

FIGURE 1.2 GRAPH OF PREDICTED FAILURE RATES FO
MODEL I, MODEL IV(.8), AND TRUE RAT

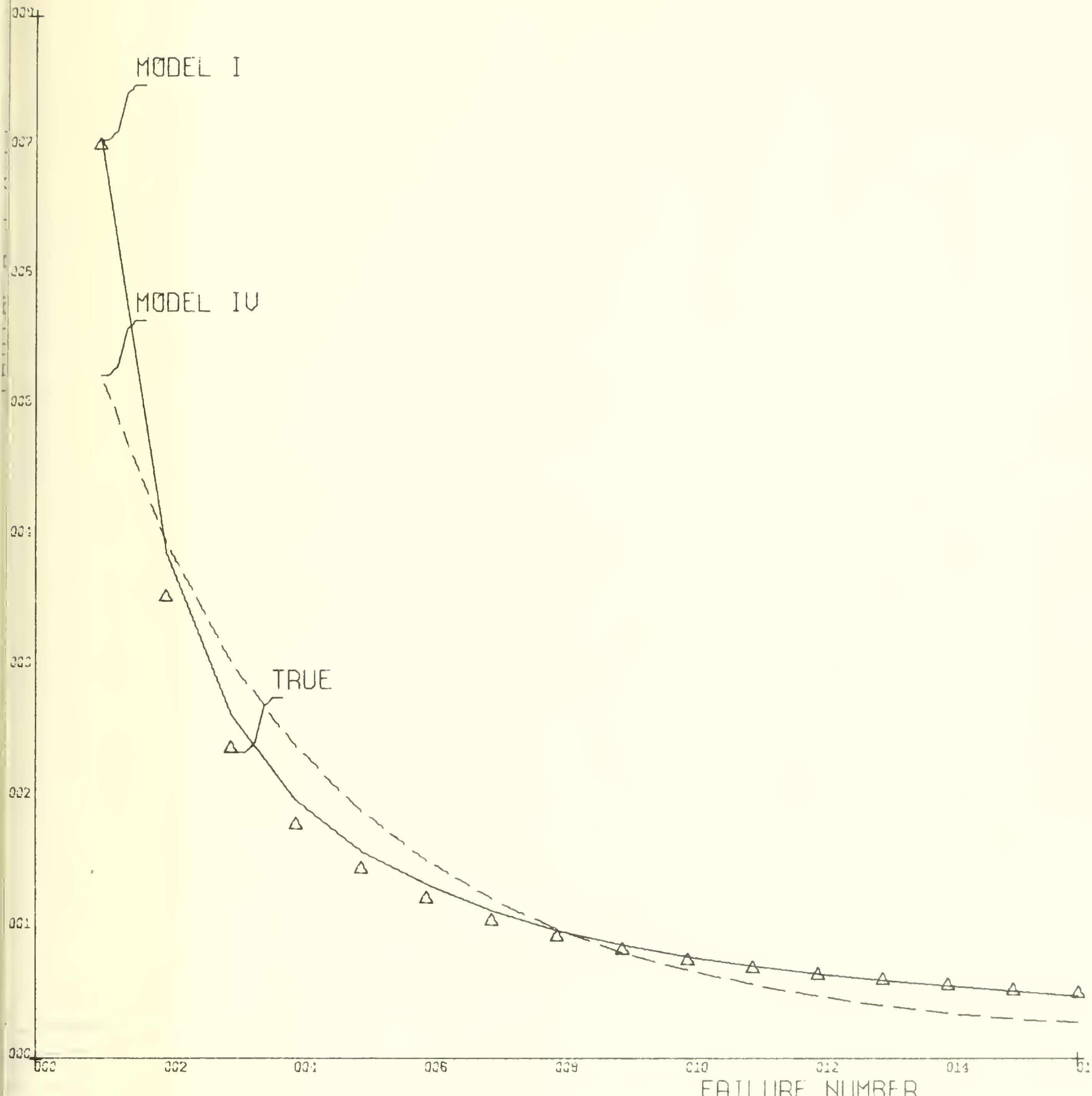

FIGURE 1.3 GRAPH OF PREDICTED FAILURE RATES FOR MODEL I, MODEL IV(.8), AND TRUE RATE

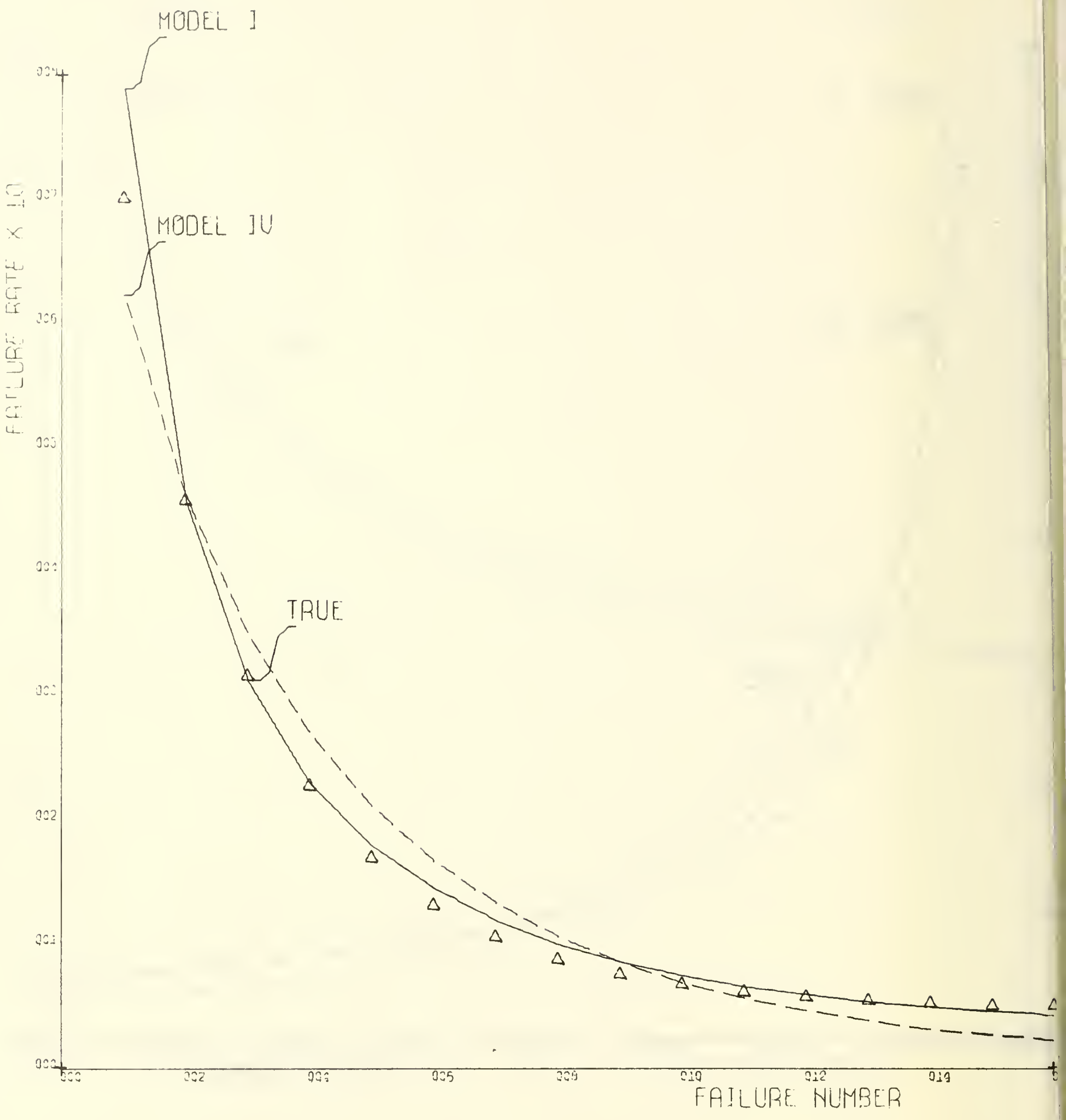

FIGURE 1.4 GRAPH OF PREDICTED FAILURE RATES F
MODEL I, MODEL IV(.8), AND TRUE RA

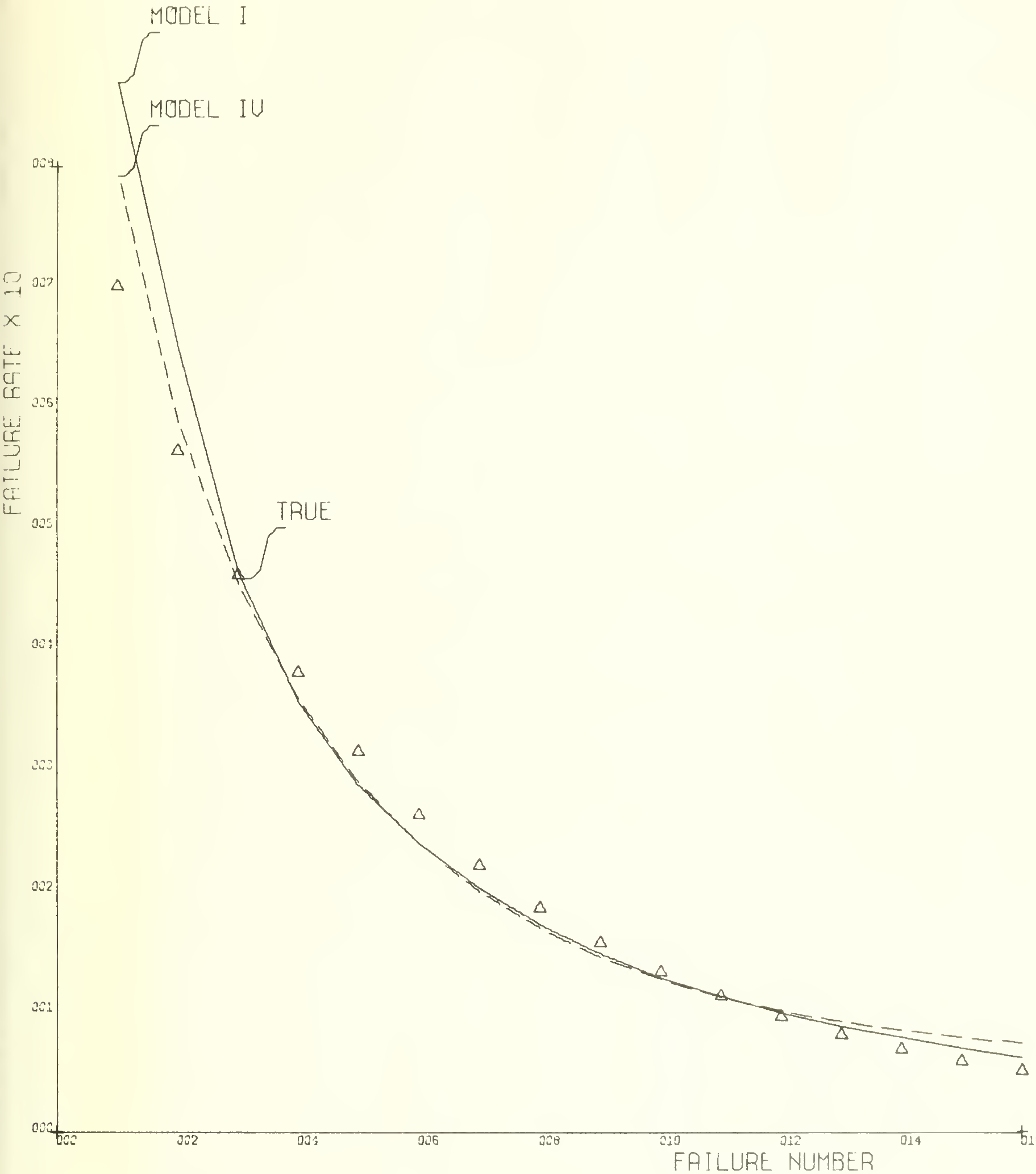

FIGURE 1.5 GRAPH OF PREDICTED FAILURE RATES FOR MODEL I, MODEL IV(.8), AND TRUE RATE

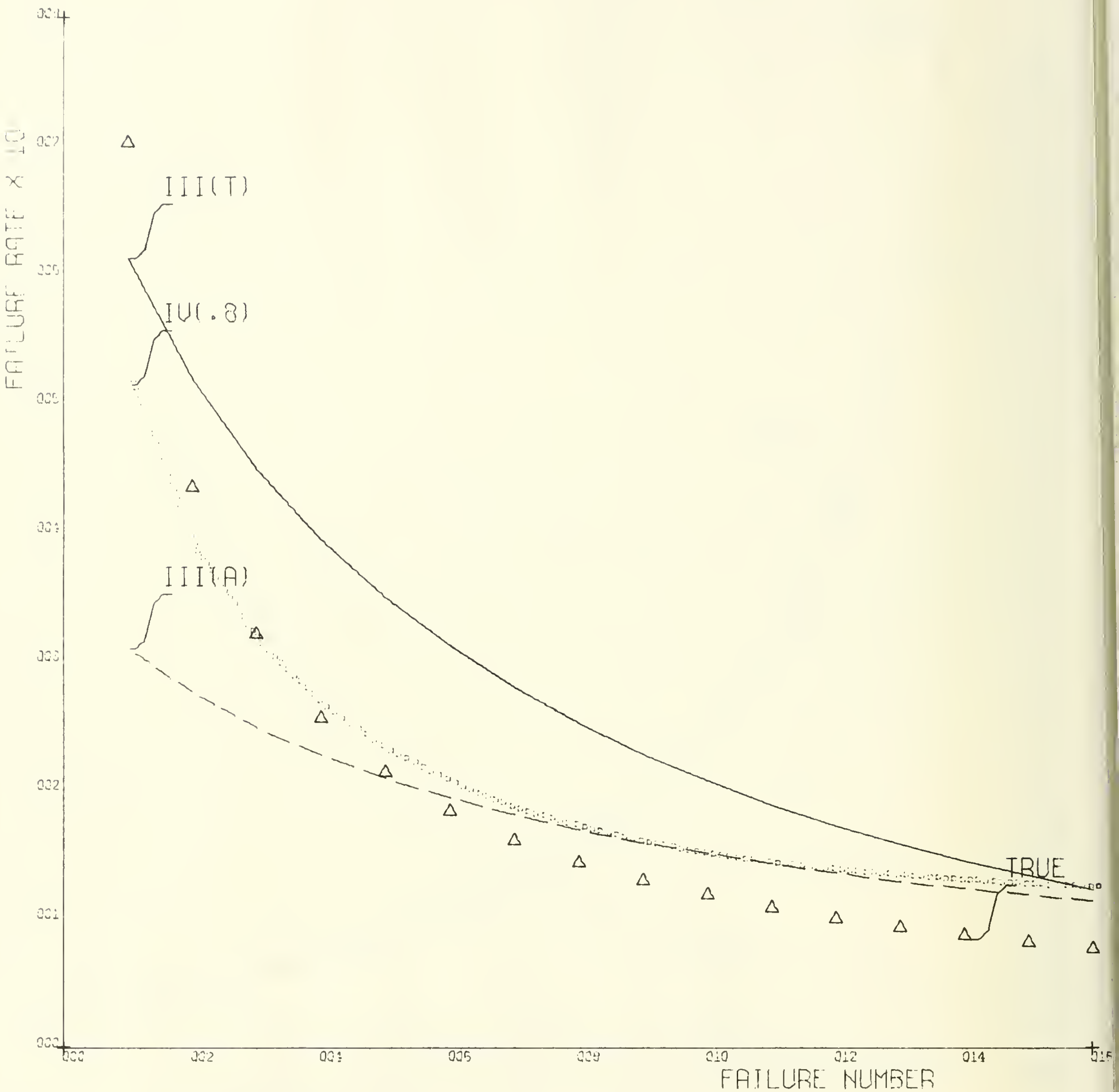

FIGURE 2.1 GRAPH OF FAILURE RATES PREDICTED BY
MODELS III(T&A), IV(.8) AND TRUE RA

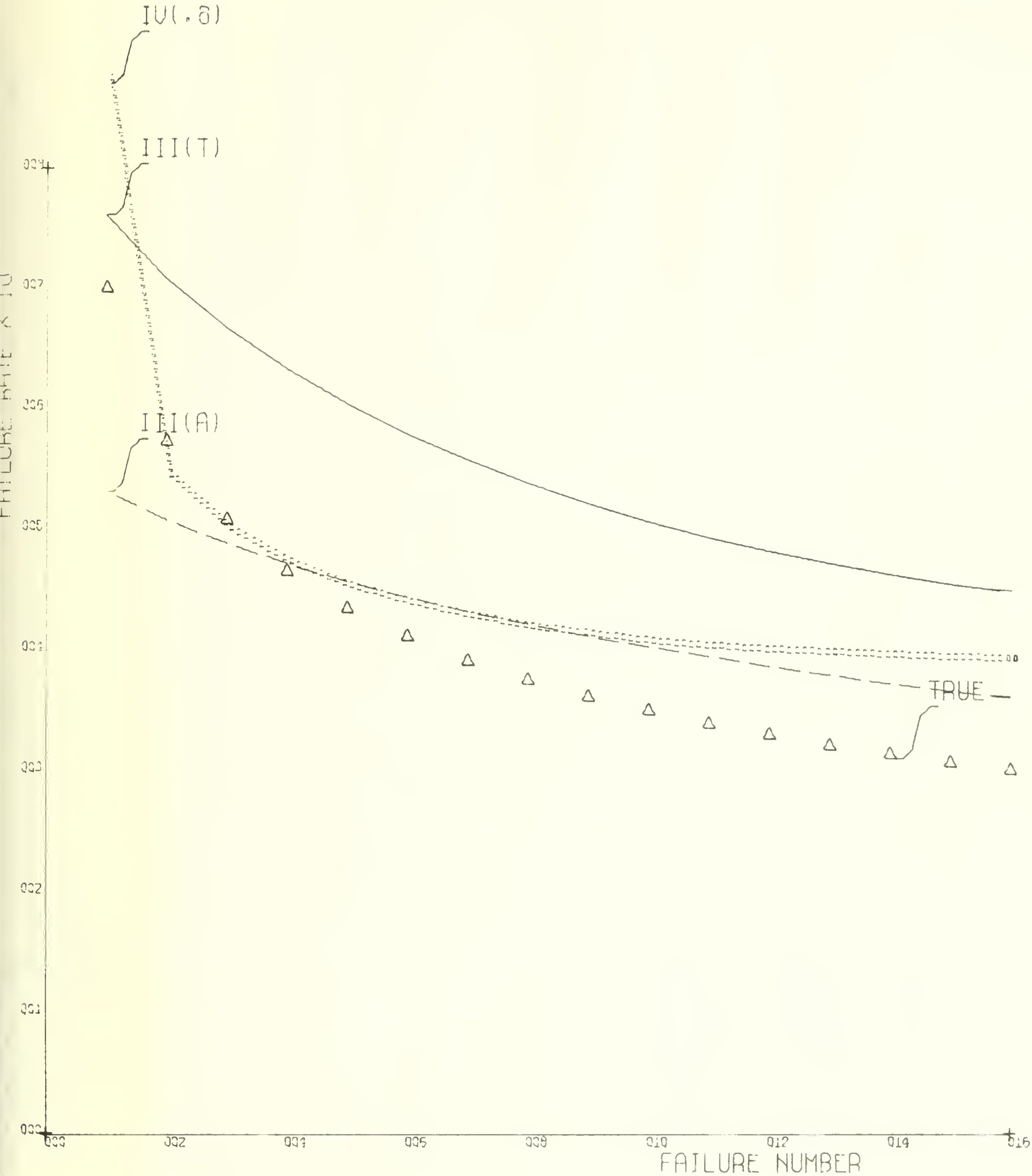

FIGURE 2.2 GRAPH OF FAILURE RATES PREDICTED BY MODELS III(T&A), IV(.8) AND TRUE RATE

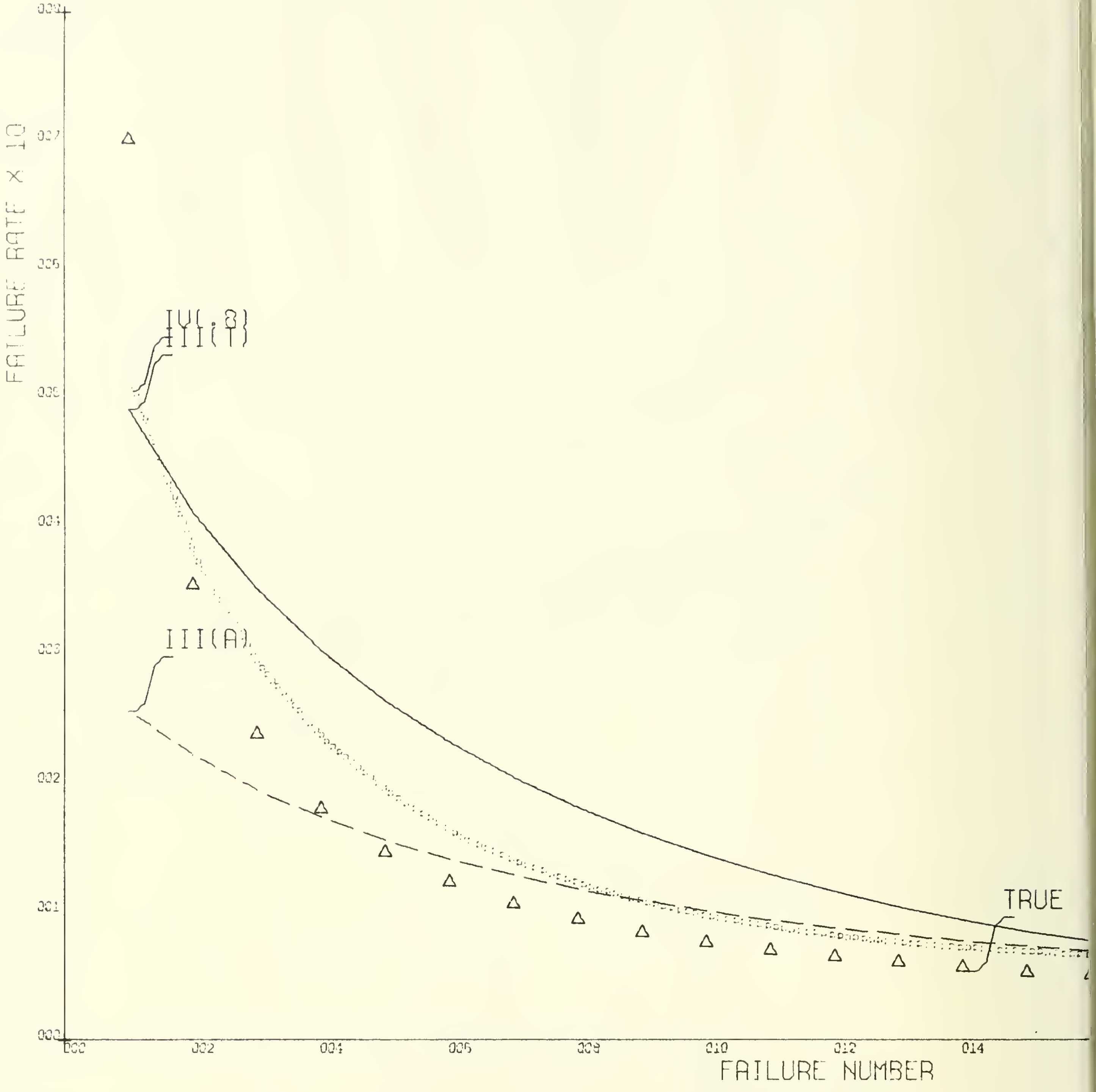

FIGURE 2.3 GRAPH OF FAILURE RATES PREDICTED B
MODELS III(T&A), IV(.8) AND TRUE F

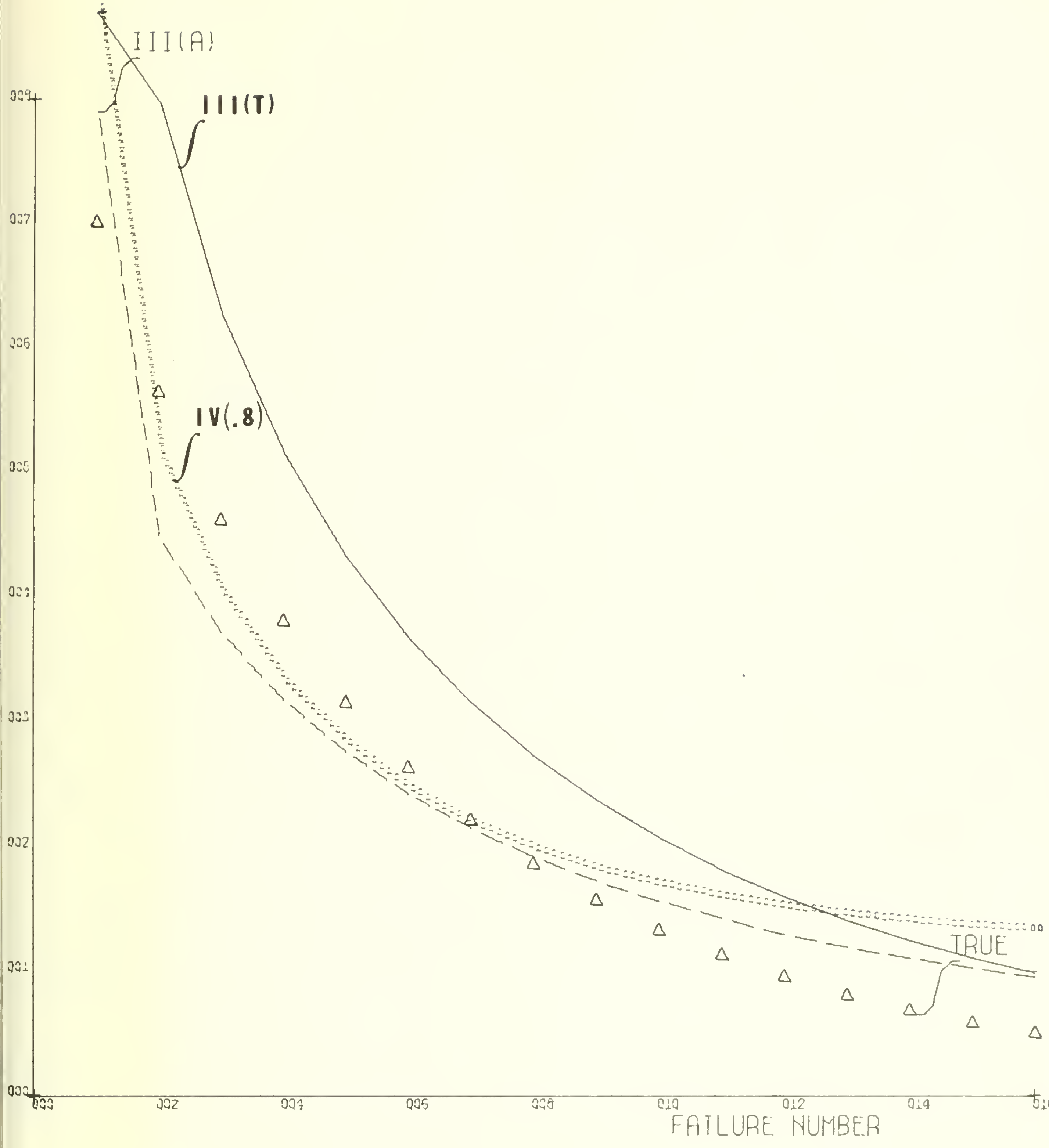

FIGURE 2.4 GRAPH OF FAILURE RATES PREDICTED BY
MODELS III(T&A), IV(.8) AND TRUE RATE

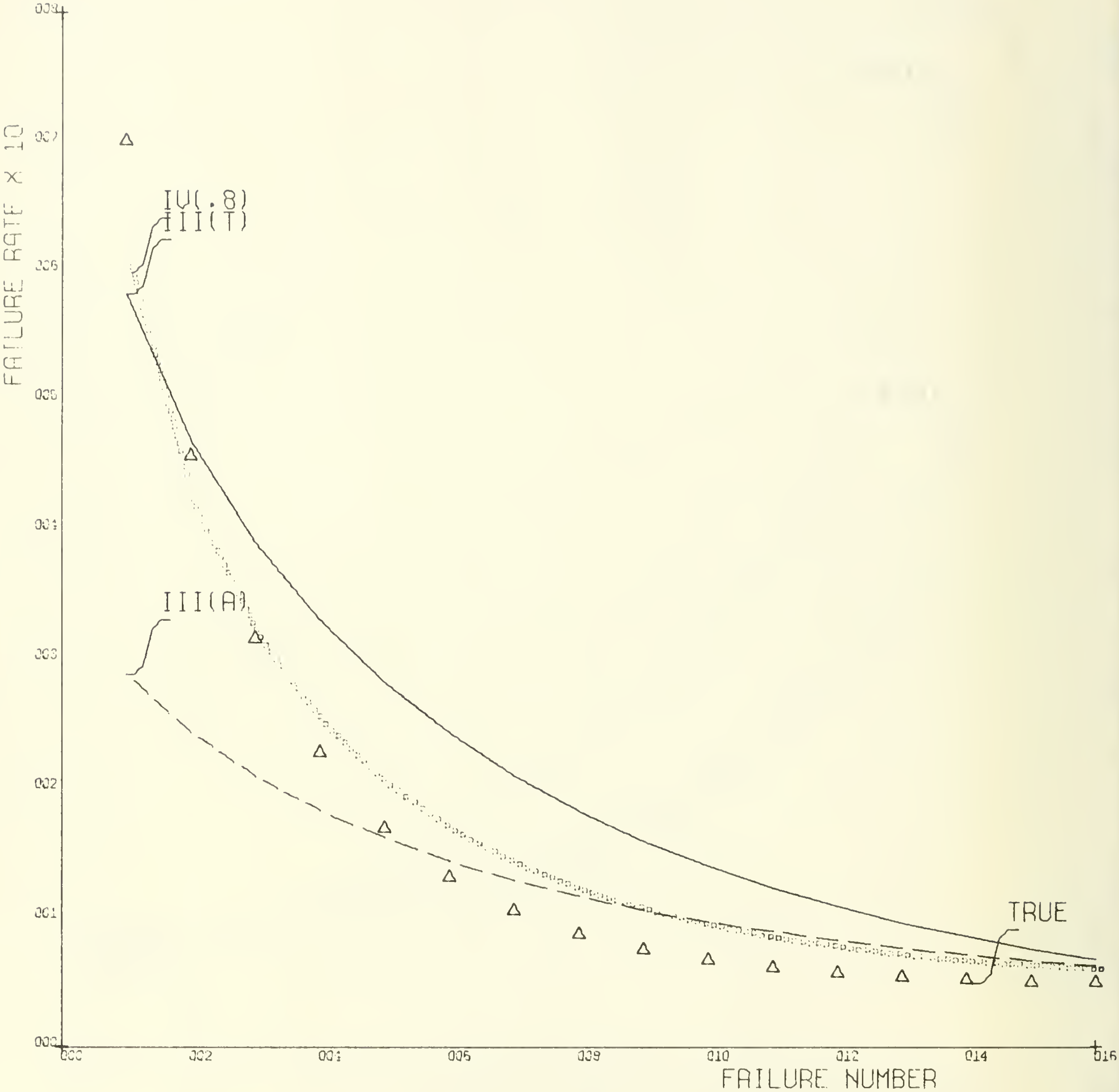

FIGURE 2.5 GRAPH OF FAILURE RATES PREDICTED BY MODELS III(T&A), IV(.8) AND TRUE RAT

References

[1] Bresenham, Jack E., "Reliability Growth Models," Stanford University Technical Report No. 74, August 1964.

[2] Chernoff, Herman and Woods, W. Max, "Reliability Growth Models - - Analysis and Applications," CEIR, Inc., file memo dated February 26, 1962.

[3] Corcoran, W. J. and Read, R. R., "Comparison of Some Reliability Growth Estimation and Prediction Schemes," UTC 2140-ITR Addendum, United Technology Center, Sunnyvale, California, June 1967.

[4] Duane, J. T., "Learning Curve Approach to Reliability Monitoring," IEEE Transactions on Aerospace, Vol. 2, No. 2, 1964, pp. 563-566.

[5] Weiss, H. K., "Estimation of Reliability Growth in A Complex System with a Poisson-type Failure," Operations Research, Vol. 4, 1956, pp. 532-545.

[6] Wolman, W., "Problems in System Reliability Analysis,: Statistical Theory of Reliability, University of Wisconsin, 1963.

```
// EXEC FORTCLG,REGION=200K
//FORT.SYSIN DD *
C
C     RELIABILITY GROWTH SIMULATION PROGRAM
C
C     ******************************************************************
C     SAMPLE DATA DECK FOLLOWS THIS PROGRAM
C     ******************************************************************
C
C     INPUT
C        DEVICE 5 "CARDS"
C        COL 1 - 4
C             KEY WORD
C             FINISH => TERMINATES PROGRAM
C                        PROGRAM ALSO TERMINATES UPON END OF DATA
C             NEW => NEW SET OF TRUE RATES FOLLOW
C                  DATA IS INPUT VIA NAMELISTS LAMDA AND TIMES
C
C                  &LAMDA NUM=NN,NOFMT=FF &END
C
C                  WHERE NN IS THE NUMBER OF LAMDAS
C                  AND FF IS THE NUMBER OF FORMAT CARDS
C                  WHICH FOLLOW AS "(" FORMAT ")"
C                  TRUE LAMDA VALUES FOLLOW FORMAT CARDS
C
C                  &TIMES  NOREP=RR,ISEED=SSSSSSSS &END
C
C                  WHERE RR = NUMBER OF REPITITIONS AND
C                  SSSSSSSS = THE SEED FOR THE SIMULATION
C
C             (THE NEXT 4 KEY WORDS CAUSE THE MODELS
C              INDICATED TO PREDICT THE TRUE LAMDAS)
C             GENERAL => MODEL I
C             WEISS => MODEL II
C             WOODS=> MODEL III
C             WOLMAN => MODEL IV  COL 41 - 50 ARE FOR BETA
C
C             RESET => RESETS ALL PRINT OPTIONS TO INITIAL CONDITIONS
C                  COL 41 - 50 = FIRST MEASURE PRINTED
C                  COL 51 - 60 = NUMBER OF SIMULATIONS
C                  COL 71 - 80 = OUTPUT DEVICE FOR MEASURES
C                  COL 61 - 70 = LAST MEASURE PRINTED
C             PRINT => PRINT OUT RESULTS TO DATE
C             A CARD WITH NO KEY WORD CAUSES THE PREDICTIONS
C                  BASED ON THE RANDOM NUMBERS GENERATED
C                  TO BE USED AS IF THEY WERE THE RESULT OF
C                  ONE OF THE MODELS
C
```

```
C
C
C      OUTPUT
C         DEVICE 6 "PRINTER"
C              RESULTS OF EACH SIMULATION
C                   USUALLY DUMMIED FOR LARGE NUMBER OF SIMULATIONS
C
C         DEVICE IN COL 71 - 80 OF RESET CARD
C              FINAL RESULTS FOR MEASURES SPECIFIED
C
C         DEVICE 9 "CARDS"
C              PREDICTED LAMDAS ARE OUTPUT FOR PLOTTING
C
C
C
C      *****************************************************************
C      THIS PROGRAM CALCULATES AND OUTPUTS OTHER QUANTITIES
C         THAN THE ONES MENTIONED ABOVE FOR FUTURE RESEARCH.
C      *****************************************************************
C
C
C
C      PROGRAMS CALLED OTHER THAN THOSE PRESENTED BELOW
C      ARE IN THE NPS PROGRAM LIBRARY
C      OF SPECIAL SIGNIFICANCE IS THE PROGRAM  "LLRANDOM"
C      WHICH GENERATES OUR RANDOM NUMBERS
C
C
       REAL*8 HOLDX(300)
       REAL*8 HOLDXX(300)
       DIMENSION FITTED(301)                                             PUB00010
       DIMENSION AMDA(301)                                               PUB00020
       REAL*4 MODEL(10)                                                  PUB00030
       DIMENSION PARM(4)                                                 PUB00040
       REAL*4 MEAS(8,6)                                                  PUB00050
       REAL*4 WOLM/'WOLM'/                                               PUB00060
       REAL*4 NEWL/'NEW '/                                               PUB00070
       REAL*4 WEIS/'WEIS'/
       REAL*4 MOOR/'MOOR'/
       REAL*4 GENE/'GENE'/
       REAL*4 FINI/'FINI'/                                               PUB00080
       REAL*4 WOOD/'WOOD'/
       REAL*4 RESE/'RESE'/,PRIN/'PRIN'/
       LOGICAL ON
       NOUT=6
       NSIM=1
       NMEAS=3
 1     CONTINUE                                                          PUB00090
```

```
      IF(NMEAS.LE.0)NMEAS=3
      IF(MSEND .LE.0)MSEND=NMEAS
      IF(NSIM.LE.0)NSIM=1
      IF(NOUT.LE.0)NOUT=6
      READ(5,100,END=999)MODEL,PARM                                     PUB00100
100   FORMAT(10A4,4F10.0)                                               PUB00110
      WRITE(6,101)MODEL                                                 PUB00120
101   FORMAT('1',//,' ',10A4,//)                                        PUB00130
      IF(MODEL(1).NE.RESE)GO TO 4
      CALL RESET
      NMEAS=PARM(1)+.5
      NSIM=PARM(2)+.5
      NOUT=PARM(3)+.5
      MSEND=PARM(4)+.5
      GO TO 1
4     CONTINUE
      IF(MODEL(1).NE.PRIN)GO TO 5
      CALL DMPOUT(NOUT)
      GO TO 1
5     CONTINUE
      IF(MODEL(1).EQ.FINI)CALL FINISH(NOUT)
      IF(MODEL(1).NE.NEWL)GO TO 8
      CALL START(AMDA,NO,ISEED)
      GO TO 1
8     CONTINUE
      DO 7 K=1,300
      HOLDX(K)=0.0D+00
      HOLDXX(K)=0.0D+00
7     CONTINUE
      ON=.TRUE.
      NEND=NO*6
      NSEED=IABS(ISEED)/2*2+1
      DO 6 K=1,NSIM
      CALL SMLT(NSEED,FITTED,ON)
      IF(MODEL(1).EQ.WOOD)CALL WOODS(AMDA,NO,FITTED)
      IF(MODEL(1).EQ.GENE)CALL GE(AMDA,NO,FITTED)
      IF(MODEL(1).EQ.WEIS)CALL WEISS(AMDA,NO,FITTED)
      IF(MODEL(1).EQ.WOLM)CALL WOLMAN(AMDA,NO,PARM(1),FITTED)           PUB00140
      DO 9 LMN=1,NEND
      HOLDX(LMN)=FITTED(LMN)+HOLDX(LMN)
      HOLDXX(LMN)=HOLDXX(LMN)+FITTED(LMN)**2
9     CONTINUE
      IF(ON)WRITE(NOUT,888)MODEL,NSIM
888   FORMAT('0',10A4,/,' SIMULATION WILL BE REPEATED',I4,' TIMES.',
     1/,' LAMDA ESTIMATES WILL NOT BE PRINTED FOR EACH SIMULATION.')
      ON=.FALSE.
6     CONTINUE
      DO 10 K=1,NEND
```

```
      FITTED(K)=HOLDX(K)/NSIM
      HOLDXX(K)=HOLDXX(K)-HOLDX(K)*FITTED(K)
      IF(NSIM.GT.1)HOLDXX(K)=HOLDXX(K)/(NSIM-1)
      IF(HOLDXX(K).LE.0.0D+00)HOLDXX(K)=0.0D+00
      HOLDX(K)=DSQRT(HOLDXX(K))
10    CONTINUE
      X=COMP(AMDA,NO,FITTED)                                            PUB00170
      WRITE(6,102)                                                      PUB00180
102   FORMAT(//,' MEASURES OF PERFORMANCE',/,'0',8('+-------------+'),  PUB00190
     1/,' ',8('|',13X,'|'),/,                                           PUB00200
     2' | SUM OF ERRS || SUM OF ABS'' || SUM OF SQRS |',                PUB00210
     3'| SUM OF RELS || SUM OF ABS'' || SUM OF SQRS ||',                PUB00220
     4' MAX ABS ERR || MAX ABS REL |',/,' ',8('|',13X,'|'),/,           PUB00230
     5' ',8('+-------------+'),/)                                       PUB00240
      DO 3 L=1,6                                                        PUB00250
      M=L                                                               PUB00260
      DO 2 K=1,8                                                        PUB00270
      MEAS(K,M)=XMEAS(K,L)                                              PUB00280
2     CONTINUE                                                          PUB00290
      WRITE(6,103)(MEAS(KK,M),KK=1,8),M                                 PUB00300
103   FORMAT(1X,8(1X,E13.6,1X),'<= EST #',I1)                           PUB00310
3     CONTINUE                                                          PUB00320
      CALL PLOTEM(AMDA,NO,FITTED,MODEL)                                 PUB00330
      DO 222 NV=NMEAS,MSEND
      CALL STORE(NV,MEAS,MODEL)
222   CONTINUE
      IPNCH=9
      WRITE(IPNCH,777)MODEL,(AMDA(K),FITTED(K),HOLDX(K),K=1,NEND)
777   FORMAT(10A4,(3E16.8))
      GO TO 1                                                           PUB00340
999   CALL FINISH(NOUT)                                                 PUB00350
      END                                                               PUB00360
      FUNCTION COMP(AMDA,NUM,FITTED)                                    PUB00370
      DIMENSION FITTED(NUM,1)                                           PUB00380
      DIMENSION AMDA(NUM,1),ERROR(50,6),REL(50,6)                       PUB00390
      COMMON/WORK/ERROR,REL                                             PUB00400
      REAL*8 SUM                                                        PUB00410
      DO 2 J=1,6                                                        PUB00420
      DO 1 I=1,NUM                                                      PUB00430
      ERROR(I,J)=AMDA(I,1)-FITTED(I,J)                                  PUB00440
      REL(I,J)=ERROR(I,J)/AMDA(I,1)                                     PUB00450
1     CONTINUE                                                          PUB00460
2     CONTINUE                                                          PUB00470
      COMP=REL(50,6)                                                    PUB00480
      RETURN                                                            PUB00490
      ENTRY XMEAS(NTYPE,NOEST)                                          PUB00500
      GO TO (10,20,30,40,50,60,70,80),NTYPE                             PUB00510
10    CONTINUE                                                          PUB00520
```

```
      SUM=0.0D+00                                                       PUB00530
      DO 11 K=1,NUM                                                     PUB00540
      SUM=SUM+ERROR(K,NOEST)                                            PUB00550
11    CONTINUE                                                          PUB00560
      XMEAS=SUM                                                         PUB00570
      RETURN                                                            PUB00580
20    CONTINUE                                                          PUB00590
      SUM=0.0D+00                                                       PUB00600
      DO 21 K=1,NUM                                                     PUB00610
      SUM=SUM+ABS(ERROR(K,NOEST))                                       PUB00620
21    CONTINUE                                                          PUB00630
      XMEAS=SUM                                                         PUB00640
      RETURN                                                            PUB00650
30    CONTINUE                                                          PUB00660
      SUM=0.0D+00                                                       PUB00670
      DO 31 K=1,NUM                                                     PUB00680
      SUM=SUM+ERROR(K,NOEST)**2                                         PUB00690
31    CONTINUE                                                          PUB00700
      XMEAS=SUM                                                         PUB00710
      RETURN                                                            PUB00720
40    CONTINUE                                                          PUB00730
      SUM=0.0D+00                                                       PUB00740
      DO 41 K=1,NUM                                                     PUB00750
      SUM=SUM+    REL(K,NOEST)                                          PUB00760
41    CONTINUE                                                          PUB00770
      XMEAS=SUM                                                         PUB00780
      RETURN                                                            PUB00790
50    CONTINUE                                                          PUB00800
      SUM=0.0D+00                                                       PUB00810
      DO 51 K=1,NUM                                                     PUB00820
      SUM=SUM+ABS(REL(K,NOEST)     )                                    PUB00830
51    CONTINUE                                                          PUB00840
      XMEAS=SUM                                                         PUB00850
      RETURN                                                            PUB00860
60    CONTINUE                                                          PUB00870
      SUM=0.0D+00                                                       PUB00880
      DO 61 K=1,NUM                                                     PUB00890
      SUM=SUM+     REL(K,NOEST)**2                                      PUB00900
61    CONTINUE                                                          PUB00910
      XMEAS=SUM                                                         PUB00920
      RETURN                                                            PUB00930
70    CONTINUE                                                          PUB00940
      SUM=0.0D+00                                                       PUB00950
      DO 71 K=1,NUM                                                     PUB00960
      IF(ABS(ERROR(K,NOEST)).GT.SUM)SUM=ABS(ERROR(K,NOEST))             PUB00970
71    CONTINUE                                                          PUB00980
      XMEAS=SUM                                                         PUB00990
      RETURN                                                            PUB01000
```

```
80    CONTINUE
      SUM=0.0D+00
      DO 81 K=1,NUM
      IF(ABS(   REL(K,NOEST)).GT.SUM)SUM=ABS(   REL(K,NOEST))
81    CONTINUE
      XMEAS=SUM
      RETURN
      END
      SUBROUTINE PLOTEM(A,NO,F,WORDS)
      DIMENSION A(NO,1),F(NO,1),WORDS(1)
      COMMON/WORK/ORDER(600)
      DIMENSION TITLE(23)
      DIMENSION ARRAY(50,2,7)
      DATA TITLE/92HPLOT OF EST NO 1-6(CURVE 1-6) AND TRUE(CURVE 7) FOR
     1                                                  /
      DIMENSION INUM(7)
      NUM=50
      NPLOT=7
      DO 1 K=1,7
      INUM(K)=NO
1     CONTINUE
      DO 3 I=1,NO
      DO 2 K=2,7
      KM1=K-1
      ARRAY(I,2,KM1)=F(I,KM1)
      ARRAY(I,1,KM1)=I
2     CONTINUE
      ARRAY(I,2,7)=A(I,1)
      ARRAY(I,1,7)=I
3     CONTINUE
      MINUS=-1
      DO 5 L=1,10
      LP13=L+13
      TITLE(LP13)=WORDS(L)
5     CONTINUE
      CALL PPLOT(NUM,NPLOT,INUM,ARRAY,5,5HLAMDA,10,10HFAILURE NO,
     1MINUS,MINUS,92,TITLE,ORDER)
      RETURN
      END
      SUBROUTINE START(AMDA,NO,IX)
      NAMELIST /LAMDA/NUM,NOFMT
      NAMELIST /TIMES/NOREP,ISEED
      DIMENSION FITTED(1)
      COMMON/WORK/FORM(600)
      COMMON/EXPTIM/NOR,T(2500)
      DIMENSION AMDA(1)
      REAL*4 BLANK(3)/'(    ','    ','    )'/
      LOGICAL ON
```

```
      FORM(1)=BLANK(1)                                                  PUB01480
      DO 1 K=2,599                                                      PUB01490
      FORM(K)=BLANK(2)                                                  PUB01500
1     CONTINUE                                                          PUB01510
      FORM(600)=BLANK(3)                                                PUB01520
      NUM=1                                                             PUB01530
      NOFMT=1                                                           PUB01540
      READ(5,LAMDA)                                                     PUB01550
      NTOT=NOFMT*20                                                     PUB01560
      READ(5,100)(FORM(I),I=2,NTOT)                                     PUB01570
100   FORMAT(20A4)                                                      PUB01580
      READ(5,FORM)(AMDA(I),I=1,NUM)                                     PUB01590
      ISEED=123456789                                                   PUB01600
      NOREP=1                                                           PUB01610
      READ(5,TIMES)                                                     PUB01620
      NCR=NOREP                                                         PUB01720
      NO=NUM                                                            PUB01730
      IX=ISEED                                                          PUB01640
      LDSEED=0
      RETURN
      ENTRY SMLT(NSEED,FITTED,ON)
      IX=NSEED
      IF(IX.EQ.LDSEED)GO TO 25
      LDSEED=IX
      N=NO*NOR
      CALL EXPON(IX    ,T,N)                                            PUB01650
      INDX=0                                                            PUB01660
      DO 2 K=1,NUM                                                      PUB01670
      DO 2 J=1,NOREP                                                    PUB01680
      INDX=INDX+1                                                       PUB01690
      T(INDX)=T(INDX)/AMDA(K)                                           PUB01700
2     CONTINUE                                                          PUB01710
      CALL EST(AMDA,NO ,T,NOR)                                          PUB01740
      IF(ON)WRITE(6,101)NOR
101   FORMAT( //,' LAMDA AND LAMDA ESTIMATES ',//,                      PUB01760
     1 ' EST #1 => TRUE, EST #2 => MLE, EST #3  => MLE(INTEGER DATA),',  PUB01770
     2' EST #4 => RMLE,',/,' EST #5 => RMLE(INTEGER DATA), EST #6 => ATRPUB01780
     3IBUTE DATA',/,' BASED ON ',I2,' REPITITIONS',//)                  PUB01790
25    CONTINUE
      DO 12 J=1,6                                                       PUB01800
      LL=J*NO                                                           PUB01810
      L=LL-NO+1                                                         PUB01820
      DO 11 K=L,LL                                                      PUB01830
      FITTED(K)=AMDA(K)                                                 PUB01840
 11   CONTINUE                                                          PUB01850
      IF(IX.EQ.LDSEED)GO TO 12
      IF(.NOT.ON) GO TO 12
      WRITE(6,105)J,(AMDA(K),K=L,LL)                                    PUB01860
```

```
105   FORMAT('OEST #',I1,' => ',(T11,8(1X,E13.6,1X)) )                    PUB01870
12    CONTINUE                                                             PUB01880
      NSEED=IX
      RETURN                                                               PUB01890
      END                                                                  PUB01900
      SUBROUTINE EST(AMDA,NUM,T,NOREP)                                     PUB01910
      DIMENSION AMDA(NUM,1),T(NOREP,1)                                     PUB01920
      DOUBLE PRECISION SUM,SUMI                                            PUB01930
      DUMV=ALOG(100.0)
      DO 1 K=1,NUM                                                         PUB01940
C     GET TBARS                                                            PUB01950
      SUM=0.0D+00                                                          PUB01960
      SUMI=0.0D+00                                                         PUB01970
      DO 2 J=1,NOREP                                                       PUB01980
      SUM=SUM+T(J,K)                                                       PUB01990
      IT=T(J,K)                                                            PUB02000
      SUMI=SUMI+IT                                                         PUB02010
2     CONTINUE                                                             PUB02020
C     GET MAXIMUM LIKELIHOOD ESTIMATORS                                    PUB02030
      V=DUMV
      IF(SUM.GT.0.0D+00 )V=NOREP/SUM
      AMDA(K,2)=V                                                          PUB02050
      AMDA(K,4)=V                                                          PUB02060
      V=DUMV
      IF(SUMI.GT.0.0D+00)V=NOREP/SUMI
      VI=1.0D+00+V
      VI=ALOG(VI)                                                          PUB02080
      AMDA(K,3)=VI                                                         PUB02090
      AMDA(K,5)=VI                                                         PUB02100
C     GET RESTRICTED MLE IF NECESSARY                                      PUB02110
      KM1=K-1                                                              PUB02120
      DO 3 I=4,5                                                           PUB02130
      IM2=I-2                                                              PUB02140
      DO 4 L=1,KM1                                                         PUB02150
      LL=K-L                                                               PUB02160
      IF(AMDA(LL,I).GE.AMDA(K,I))GO TO 5                                   PUB02170
4     CONTINUE                                                             PUB02180
      LL=0                                                                 PUB02190
5     CONTINUE                                                             PUB02200
      LL=LL+1                                                              PUB02210
      IF(LL.GE.K)GO TO 3                                                   PUB02220
      SUM=0.0D+00                                                          PUB02230
      DO 6 L=LL,K                                                          PUB02240
      TEST=AMDA(L,IM2)
      IF(TEST.LT.0.0)TEST=DUMV
      SUM=SUM+1.0D+00/TEST
6     CONTINUE                                                             PUB02260
      LEN=K-LL+1                                                           PUB02270
```

```
      V=LEN/SUM                                                         PUB02280
      IF(I.EQ.5)V=ALOG(1.0+V)                                           PUB02290
      DO 7 L=LL,K                                                       PUB02300
      AMDA(L,I)=V                                                       PUB02310
 7    CONTINUE                                                          PUB02320
 3    CONTINUE                                                          PUB02330
C     ATTRIBUTE DATA                                                    PUB02340
      SUMI=0.0D+00                                                      PUB02350
      DO 8 J=1,NOREP                                                    PUB02360
      IT=T(J,K)                                                         PUB02370
      IF(IT.GE.1)SUMI=SUMI+1.0D+00                                      PUB02380
 8    CONTINUE                                                          PUB02390
      IF(SUMI.EQ.0.0D+00)SUMI=.01D+00                                   PUB02400
      IF(SUMI.GE.NOREP)SUMI=SUMI-1.0D-02
      PS=SUMI/NOREP                                                     PUB02410
      AMDA(K,6)= -ALOG(PS)                                              PUB02420
 1    CONTINUE                                                          PUB02430
      RETURN                                                            PUB02440
      END                                                               PUB02450
      SUBROUTINE FINISH(NOUT)                                           PUB02460
      CALL DMPOUT(NOUT)
      WRITE(6,100)                                                      PUB02470
      STOP                                                              PUB02480
 100  FORMAT('1PROGRAM TERMINATING DO TO "FINISH" COMMAND OR "/*"')     PUB02490
      END                                                               PUB02500
      SUBROUTINE WOLMAN(AMDA,NO,B,FITTED)                               PUB02510
      EXTERNAL RQF                                                      PUB02520
      COMMON/EXPTIM/NOR,T(2500)                                         PUB02530
      COMMON/WORK/W(600)                                                PUB02540
      DIMENSION FITTED(NO,1)                                            PUB02550
      REAL*8 R,Q                                                        PUB02560
      DIMENSION AMDA(NO,1),X(50),Y(50)                                  PUB02570
      EQUIVALENCE (X(1),R),(X(3),Q)                                     PUB02580
      DUM=ALOG(1.0E+30)
      N=NO                                                              PUB02590
      BI=B                                                              PUB02600
      WRITE(6,201)BI                                                    PUB02610
 201  FORMAT(//,' WOLMAN MODEL ESTIMATES, LAMDA(I) = -LN(1-R-Q*(',      PUB02620
     1E13.6,')**(I-1))',//,' EST #2 THROUGH #6 ARE FROM REGRESSION',    PUB02630
     2' ON LAMDA ESTIMATES #2 THROUGH #6,',/   ,                        PUB02640
     3 ' EST #1 IS A MLE BASED ON LAMDA ESTIMATE #3',//)                PUB02650
      KSTART=2                                                          PUB02660
C     REGRESSION                                                        PUB02670
      DO 1 K=KSTART,6                                                   PUB02680
      DO 2 L=1,N                                                        PUB02690
      Y(L)=1.0-EXP(-AMDA(L,K))                                          PUB02700
      X(L)=BI**(L-1)                                                    PUB02710
 2    CONTINUE                                                          PUB02720
```

```
      W(600)=5.0                                                        PUB02730
      CALL FIT(Y,X,RR,QQ,N)                                             PUB02740
      DO 3 L=1,N                                                        PUB02750
      PONENT=1.0-RR-QQ*X(L)
      IF(PONENT.LE.1.E-30)GO TO 100
      FITTED(L,K)=-ALOG(PONENT)
      GO TO 3
100   CONTINUE
      FITTED(L,K)=DUM
3     CONTINUE                                                          PUB02770
      R=RR                                                              PUB02780
      Q=QQ                                                              PUB02790
      WRITE(6,202)R,Q,K,(FITTED(L,K),L=1,N)                             PUB02800
202   FORMAT('OR = ',E13.6,', Q = ',E13.6,                              PUB02810
     1/,' EST #',I1,' => ',(T11,8(1X,E13.6,1X))   )                     PUB02820
1     CONTINUE                                                          PUB02830
C     MAXIMUM LIKELIHOOD ESTIMATOR BASED ON ATTRIBUTE DATA              PUB02840
      C=RQFI(BI,N)                                                      PUB02850
      CALL ZSYSTM(RQF,1.0D-04,4,2,X,100,W,IER)                          PUB02860
      IF(IER.NE.0)GO TO 50                                              PUB02870
      DO 12 L=1,N                                                       PUB02880
      PONENT= 1.0-X(1)-X(3)*BI**(L-1)
      IF(PONENT.LT.1.0E-30)GO TO 101
      FITTED(L,1)=-ALOG(PONENT)
      GO TO 12
101   CONTINUE
      FITTED(L,1)=DUM
12    CONTINUE                                                          PUB02900
      K=1                                                               PUB02910
      WRITE(6,202)R,Q,K,(FITTED(L,K),L=1,N)                             PUB02920
      GO TO 11                                                          PUB02930
50    CONTINUE                                                          PUB02940
      DO 4 L=1,N                                                        PUB02950
      FITTED(L,1)=0.0                                                   PUB02960
4     CONTINUE                                                          PUB02970
11    CONTINUE                                                          PUB02980
      RETURN                                                            PUB02990
      END                                                               PUB03000
      SUBROUTINE FIT(Y,X,A,B,N)                                         PUB03010
      COMMON/WORK/YY(600)                                               PUB03020
      DIMENSION Y(1),X(1)                                               PUB03030
      DOUBLE PRECISION SUMX,SUMY,SUMXY,SUMXX                            PUB03040
      DATA IENTER/0/                                                    PUB03050
      IENTER=IENTER+1                                                   PUB03060
      M=YY(600)+.5                                                      PUB03070
      NN=N                                                              PUB03080
      SUMX=0.0                                                          PUB03090
      SUMY=0.0                                                          PUB03100
```

```
      SUMXY=0.0                                                         PUB03110
      SUMXX=0.0                                                         PUB03120
901   FORMAT('1PLOT OF  DATA TO BE FITTED',//)                          PUB03150
51    CONTINUE                                                          PUB03170
      DO 1 K=1,N                                                        PUB03180
      SUMX=SUMX+X(K)                                                    PUB03190
      SUMY=SUMY+Y(K)                                                    PUB03200
      SUMXY=SUMXY+X(K)*Y(K)                                             PUB03210
      SUMXX=SUMXX+X(K)*X(K)                                             PUB03220
1     CONTINUE                                                          PUB03230
      XBAR=SUMX/N                                                       PUB03240
      BNUM=SUMXY-XBAR*SUMY                                              PUB03250
      BDEN=SUMXX-XBAR*SUMX                                              PUB03260
      B=BNUM/BDEN                                                       PUB03270
      A=SUMY/N-B*XBAR                                                   PUB03280
      DO 99 K=1,N                                                       PUB03290
      YY(K)=A+B*X(K)                                                    PUB03300
99    CONTINUE                                                          PUB03310
      IF(IENTER.EQ.M)IENTER=0                                           PUB03330
      RETURN                                                            PUB03350
      END                                                               PUB03360
      REAL FUNCTION RQFI*8(B,N)                                         PUB03370
      REAL*8 SUM,BB,X(2),TT(50)                                         PUB03380
      REAL*8 R,Q                                                        PUB03390
      REAL*8 RQF                                                        PUB03400
      COMMON/EXPTIM/NOR,T(2500)                                         PUB03410
      BB=B                                                              PUB03420
      NN=N                                                              PUB03430
      INDX=0                                                            PUB03440
      DO 1 L=1,NN                                                       PUB03450
      SUM=0.0D+00                                                       PUB03460
      DO 4 K=1,NOR                                                      PUB03470
      INDX=INDX+1                                                       PUB03480
      IT=T(INDX)                                                        PUB03490
      SUM=SUM+IT                                                        PUB03500
4     CONTINUE                                                          PUB03510
      TT(L)=SUM/NOR                                                     PUB03520
1     CONTINUE                                                          PUB03530
      RQFI=NOR                                                          PUB03540
      RETURN                                                            PUB03550
      ENTRY RQF(X,K)                                                    PUB03560
      SUM=0.0D+00                                                       PUB03570
      R=X(1)                                                            PUB03580
      Q=X(2)                                                            PUB03590
      GO TO (10,20),K                                                   PUB03600
10    CONTINUE                                                          PUB03610
      DO 2 L=1,NN                                                       PUB03620
      SUM=SUM+TT(L)/(1.0D+00-R-Q*BB**(L-1))                             PUB03630
```

```
      SUM=SUM-1.0D+00/(R+Q*BB**(L-1))
2     CONTINUE
      RQF=SUM
      RETURN
20    CONTINUE
      DO 3 L=1,N
      SUM=SUM+TT(L)/(1.0D+00-R-Q*BB**(L-1))*BB**(L-1)
      SUM=SUM-1.0D+00/(R+Q*BB**(L-1))*BB**(L-1)
3     CONTINUE
      RQF=SUM
      RETURN
      END
      SUBROUTINE GE(AMDA,NO,FITTED)
      DIMENSION AMDA(NO,1),FITTED(NO,1),X(50),Y(50)
      COMMON/WORK/W(600)
      COMMON/EXPTIM/NOR,T(2500)
      REAL*8 CUMT,SUMREP
      WRITE(6,201)
201   FORMAT(//,' GENERAL ELECTRIC MODEL ESTIMATES, LAMDA(T) = ',
     1'C*T**(-ALPHA)',//,' ALL ESTIMATES ARE FROM REGRESSIONS ON',
     2' CORRESPONDING LAMDA ESTIMATES',//)
      W(600)=6.0
      DO 3 NOEST=1,6
      CUMT=0.0D+00
      INDX=0
      DO 1 K=1,NO
      SUMREP=0.0D+00
      DO 2 J=1,NOR
      INDX=INDX+1
      TI=T(INDX)
      IT=TI+1
      GO TO (10,10,20,10,20,30),NOEST
10    CONTINUE
      SUMREP=SUMREP+TI
      GO TO 2
20    CONTINUE
      SUMREP=SUMREP+IT
      GO TO 2
30    CONTINUE
      SUMREP=SUMREP+NOR
      J=NOR
2     CONTINUE
      CUMT=CUMT+SUMREP
      XK=CUMT
      X(K)=-ALOG(XK)
      IF(AMDA(K,NOEST).LE.1.0E-60)GO TO 300
      Y(K)=ALOG(AMDA(K,NOEST))
1     CONTINUE
```

```
      CALL FIT(Y,X,AK,ALPHA,NO)
      AK=EXP(AK)
      DO 4 K=1,NO
      FITTED(K,NOEST)=AK*EXP(X(K)*ALPHA)
4     CONTINUE
      WRITE(6,202)AK,ALPHA,NOEST,(FITTED(L,NOEST),L=1,NO)
202   FORMAT('OC = ',E13.6,', ALPHA = ',E13.6,
     1/,' EST #',I1, ' => ',(T11,8(1X,E13.6,1X)))
      GO TO 3
300   WRITE(6,203)NOEST,K,AMDA(K,NOEST)
203   FORMAT('OERROR IN EST #',I1,/,' VALUE FOR FAILURE NO ',
     1I2,' IS ',E13.6)
      DO 31 K=1,NO
      FITTED(K,NOEST)=0.0
31    CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE WEISS(AMDA,NO,FITTED)
      EXTERNAL DBIFDI
      REAL*8 BB
      DIMENSION AMDA(NO,1),FITTED(NO,1),X(50),Y(50)
      COMMON/WORK/W(600)
      W(600)=5.0
      WRITE(6,201)
201   FORMAT(//,' WEISS MODEL ESTIMATES, LAMDA(I) = ',
     1'1.0/A + B/(A*I)',//,' EST #2 THROUGH #6 ARE FROM REGRESSION',
     2' ON LAMDA ESTIMATES #2 THROUGH #6,',/,
     3' EST #1 IS A MLE BASED ON LAMDA ESTIMATE #2',//)
      DO 2 NOEST=2,6
      DO 1 K=1,NO
      Y(K)=AMDA(K,NOEST)
      X(K)=1.0/K
1     CONTINUE
      CALL FIT(Y,X,A,B,NO)
      DO 3 K=1,NO
      FITTED(K,NOEST)=A+B*X(K)
3     CONTINUE
      A=1.0/A
      B=B*A
      WRITE(6,202)A,B,NOEST,(FITTED(L,NOEST),L=1,NO)
202   FORMAT('OA = ',E13.6,', B = ',E13.6,/,
     1' EST #',I1, ' => ',(T11,8(1X,E13.6,1X)))
2     CONTINUE
      C=BFI(NO)
      B=0.0
      LOOPS=50
      DO 10 I=1,LOOPS
```

```
      DELTB=BIFDI(B)/DERIV(B)
      B=B-DELTB
      IF(ABS(DELTB/B).LT.1.0E-04)GO TO 11
10    CONTINUE
      GO TO 12
11    CONTINUE
      A=HAY(B)
      DO 4 K=1,NO
      FITTED(K,1)=1.0/A+B/A*X(K)
4     CONTINUE
      NOEST=1
      WRITE(6,202)A,B,NOEST,(FITTED(L,NOEST),L=1,NO)
      RETURN
12    CONTINUE
      BB=B
      I00=100
      CALL ZREAL1(DBIFDI,1.D-04,0.D+00,0.D+00,4,1,BB,I00,IER)
      B=BB
      IF(IER.EQ.0)GO TO 11
      WRITE(6,204)
204   FORMAT('0ERROR IN ESTIMATE #6. SOLUTION TO B FAILED TO CONVERGE.')
      DO 31 K=1,NO
      FITTED(K,NOEST)=0.0
31    CONTINUE
      RETURN
      END
      FUNCTION BFI(NO)
      COMMON/EXPTIM/NOR,T(2500)
      REAL*8 SUM,SUMT,SUMTDI
      REAL*8 SUMI
      SUMT=0.0D+00
      SUMTDI=SUMT
      INDX=0
      DO 2 K=1,NO
      SUM=0.0D+00
      DO 1 J=1,NOR
      INDX=INDX+1
      SUM=SUM+T(INDX)
1     CONTINUE
      SUM=SUM/NOR
      SUMT=SUMT+SUM
      SUMTDI=SUMTDI+SUM/K
2     CONTINUE
      BFI=SUMT/SUMTDI
      RETURN
      ENTRY BIFDI(B)
      SUM=0.0D+00
       SUMI=0.0D+00
```

```
      DO 3 K=1,NO
      SUM=SUM+1.0/(B+K)
      SUMI=SUMI+K/(B+K)
3     CONTINUE
      BIFDI=SUMT*SUM-SUMTDI*SUMI
      RETURN
      ENTRY DERIV(B)
      SUM=0.0D+00
       SUMI=0.0D+00
      DO 4 K=1,NO
      SUM=SUM+1.0/(B+K)**2
      SUMI=SUMI+K/(B+K)**2
4     CONTINUE
      DERIV=SUMTDI*SUMI-SUMT*SUM
      RETURN
      ENTRY HAY(B)
      HAY=(B*SUMTDI+SUMT)/NO
      RETURN
      END
      REAL FUNCTION DBIFDI*8(BB)
      REAL*8 BB
      B=BB
      DBIFDI=BIFDI(B)
      RETURN
      END
      SUBROUTINE WOODS(AMDA,NO,FITTED)
      DIMENSION AMDA(NO,1),FITTED(NO,1),X(50),Y(50)
      COMMON/WORK/W(600)
      COMMON/EXPTIM/NOR,T(2500)
      TEST=ALOG(.999999)
      WRITE(6,201)
201   FORMAT(//,' WOODS-CHERNOFF MODEL ESTIMATES, LAMDA(I) = -',
     1'LN(1.0 - EXP(-ALPHA - BETA**(I-1)))',//,' ALL ESTIMATES',
     2' ARE FROM REGRESSIONS ON CORRESPONDING LAMDA ESITMATES',//)
      W(600)=6.0
      DO 3 NOEST=1,6
      DO 1 K=1,NO
      X(K)=K-1
      IF(AMDA(K,NOEST).LE.-TEST  )GO TO 300
      Y(K)=-ALOG(1.0-EXP(-AMDA(K,NOEST)))
1     CONTINUE
      CALL FIT(Y,X,ALPHA,BETA,NO)
      DO 2 K=1,NO
      ONENT=-(ALPHA+BETA*X(K))
      IF(ONENT.GE.TEST)ONENT=TEST
      FITTED(K,NOEST)=-ALOG(1.0-EXP(ONENT))
2     CONTINUE
      WRITE(6,202)ALPHA,BETA,NOEST,(FITTED(L,NOEST),L=1,NO)
```

```
202   FORMAT('OALPHA = ',E13.6,', BETA = ',E13.6,/,
     1' EST #',I1,' => ',(T11,8(1X,E13.6,1X)))
      GO TO 3
300   WRITE(6,203)NOEST,K,AMDA(K,NOEST)
203   FORMAT('OERROR IN EST #',I1,/,' VALUE FOR FAILURE NO ',
     1I2,' IS ',E13.6)
      DO 31 K=1,NO
      FITTED(K,NOEST)=0.0
31    CONTINUE
3     CONTINUE
      RETURN
      END
      SUBROUTINE UERTST(NAME)
      LOGICAL*1 NAME(6)
      RETURN
      END
      SUBROUTINE STORE(NMEAS,VALUES,TITLE)
      DIMENSION TITLE(1),VALUES(8,1),OUTPUT(10,60)
      DATA NCOL/0/
      NCOL=NCOL+1
      GO TO 1
      ENTRY RESET
      NCOL=0
      RETURN
1     CONTINUE
      DO 2 K=1,3
      OUTPUT(K,NCOL)=TITLE(K)
2     CONTINUE
      OUTPUT(4,NCOL)=NMEAS
      DO 3 K=5,10
      KM4=K-4
      OUTPUT(K,NCOL)=VALUES(NMEAS,KM4)
3     CONTINUE
      RETURN
      ENTRY DMPOUT(NOUT)
      IF(NCOL.EQ.0)RETURN
      WRITE(NOUT,101)        (L,L=1,6)
101   FORMAT('1',//,' COMPARISON OF MODELS  ',
     1//,' MODEL          | MEASURE',6(' |    EST #',I1,3X),/,
     2'+','____________',6('______________'),'__________')
      DO 5 K=1,NCOL
      WRITE(NOUT,102)(OUTPUT(L,K),L=1,10)
102   FORMAT(' ',3A4,' |',F6.0,2X,6(' |',E13.6))
5     CONTINUE
      WRITE(NOUT,101)(L,L=2,22,4)
      DO 6 K=1,NCOL
      ITEST=(K   +1)/4*4-(K   +1)
      IF(ITEST.NE.0) WRITE(NOUT,102) OUTPUT(1,K),OUTPUT(2,K),
```

```
      1OUTPUT(3,K),OUTPUT(4,K),OUTPUT(6,K),OUTPUT(10,K)
 6     CONTINUE
       WRITE(NOUT,213)
 213   FORMAT(1H1)
       RETURN
       END
//GO.FT08F001 DD SYSOUT=A
//GO.FT09F001 DD SYSOUT=B
//GO.SYSIN DD *
RESET                                                 1.0       1.        8.0       8.0
NEW LAMDAS    WEISS
 &LAMDA
 NUM=16,NOFMT=1
 &END
(5F10.0)
.7        .353333333.237777778.18      .145333333
.122222222.105714286.093333333.083703704.076
.069696970.064444444.06      .056190476.052888888
.05
 &TIMES
 NUREP=10,ISEED=14024913
 &END
(B+I)/(A*I)    B=104, A=150
GENERAL ELECTRIC
WEISS
WOODS
WOLMAN (.10)                                         .1
WOLMAN (.20)                                         .2
WOLMAN (.30)                                         .3
PRINT RESULTS
RESET                                                 1.0       1.        8.0       8.0
WOLMAN (.40)                                         .4
WOLMAN (.50)                                         .5
WOLMAN (.60)                                         .6
WOLMAN (.70)                                         .7
WOLMAN (.80)                                         .8
WOLMAN (.90)                                         .9
PRINT RESULTS
FINISH
```

```
C
C     PROGRAM TO GENERATE MODEL I DATA  PROGRAMMED ON HP 9810
C     DEFINITION OF MODEL IS A FIXED POINT EQUATION
C     WHICH WAS SOLVED RECURSIVLY
C
C
C     PROGRAM TO GENERATE MODEL II DATA PROGRAMMED ON HP 9810
C     GENERATED IN STRAIGHT FORWARD MANNER FROM DEFINITION
C
C
C     PROGRAM TO GENERATE MODEL III DATA
C
      REAL*4 LAMDA(16)
      ALPHA=-ALOG(1.0-EXP(-.7))
      BETA=ALOG((1.0-EXP(-.7))/(1.0-EXP(-.05)))
      BETA=BETA/15
      DO 1 K=1,16
      KM1=K-1
      LAMDA(K)=-ALOG(1.0-EXP(-ALPHA-BETA*KM1))
 1    CONTINUE
      WRITE(6,101)ALPHA,BETA,(LAMDA(I),I=1,16)
 101  FORMAT('OALPHA=',E15.7,', BETA=',E15.7,
     1/,(5E18.9,/))
      WRITE(7,102)LAMDA
 102  FORMAT(5F10.9)
      STOP
      END
C
C     PROGRAM TO GENERATE MODEL IV DATA
C
      REAL*4 LAMDA(16)
      B=.7
      RPQ=1.0-EXP(-.7)
      RPQ15=1.0-EXP(-.05)
      Q15MQ=RPQ15-RPQ
      Q=Q15MQ/(B**15-1.0)
      R=RPQ-Q
      DO 1 K=1,16
      KM1=K-1
      LAMDA(K)=-ALOG(1.0-R-Q*B**KM1)
 1    CONTINUE
      WRITE(6,101)R,Q,B
 101  FORMAT('OLAMDA(I)=-ALOG(1.0-(',E15.7,')-(',
     1E15.7,')*(',E15.7,')**(I-1)')
      WRITE(7,102)(  LAMDA(I),I=1,16)
 102  FORMAT(5F10.9)
      STOP
      END
```

DISTRIBUTION LIST

Defense Documentation Center (DDC) Cameron Station Alexandria, Virginia 22314	2
Library Naval Postgraduate School Monterey, California 93940	2
Department of Mathematics Naval Postgraduate School Monterey, California 93940	1
Professor Toke Jayachandran Code 53Jy Naval Postgraduate School Monterey, California 93940	15
Dean of Research Administration Code 023 Naval Postgraduate School Monterey, California 93940	2
Dean W. Max Woods Code 500/024 Naval Postgraduate School Monterey, California 93940	1
LT L. R. Moore, III Code 53Ml Naval Postgraduate School Monterey, California 93940	10